

## CLAIM LISTING

This listing of claims will replace all prior versions, and listings of claims in the application:

### IN THE CLAIMS

1. (Previously Presented) A computer-automated method for electronic design specification comprising:
  - obtaining a target architecture;
  - specifying at least one resource or functionality for the target architecture using at least one construct in a Resource Description Language (RDL) to produce an RDL description;
  - generating an intermediate graph representation from the RDL description;
  - traversing the intermediate graph representation; and
  - invoking results of the traversing step using a high-level synthesis compiler to generate a functionality graph and resource graphs for the target architecture.
- 2-20. (Cancelled)
21. (Original) The method of claim 1 wherein:
  - a resource is used to specify an architecture and a plurality of functionalities.
22. (Original) The method of claim 1 wherein:
  - a unit is used to specify a hardware structure comprising a hierarchical representation of one or more hardware structure.
23. (Original) The method of claim 1 wherein:
  - a UNITDEF value defines or describes a hierarchy of a unit.
24. (Original) The method of claim 1 wherein:

a RESOURCEDEF value defines a resource among a set of functionality or associated property.

25. (Original) The method of claim 1 wherein:

a RCONNECT value denotes a connection between an origin resource and a destination resource via a connecting resource.

26. (Original) The method of claim 1 wherein:

a USES value indicates one or more resource used by a particular resource, the USES value defining at least one virtual resource for building at least one physical resource in an architecture.

27. (Original) The method of claim 1 wherein:

a FUNCTIONALITY value specifies a set of one or more basic operator to provide a functionality.

28. (Original) The method of claim 1 wherein:

a FUNCTIONALITYDEF value defines a composition of a new functionality.

29. (Original) The method of claim 1 wherein:

a DCONNECT value connects a plurality of basic operators while constructing a new functionality.

30. (Original) The method of claim 1 wherein:

an INPUT value specifies one or more input node for constructing a new functionality .

31. (Original) The method of claim 1 wherein:

an OUTPUT value specifies one or more output node for constructing a new functionality.

32. (Original) The method of claim 1 wherein:

an OPT\_INPUT value specifies one or more optional input node while constructing a new functionality.

33. (Original) The method of claim 1 wherein:

an if value specifies an arbitrarily complex connection between a plurality of resources in conjunction with using a for value.

34. (Original) The method of claim 1 wherein:

a for value specifies an arbitrarily complex connection between a plurality of resources in an architecture.

35. (Original) The method of claim 1 wherein:

at least one operator in a resource design language (RDL) specifies a hardware and a processing of the hardware.

36. (Original) The method of claim 32 wherein:

a hierarchy traversal operator (->) specifies a unit or resource embedded within one or more units by specifying a chain of units hierarchically with the -> operator denoting a child-parent relationship in a hierarchy.

37. (Original) The method of claim 32 wherein:

an array operator ([ ]) specifies an array or collection of one or more resource or unit.

38. (Original) The method of claim 32 wherein:

a comment operator (//) inserts one or more comment in an architecture file.

39. (Original) The method of claim 32 wherein:

operators +, -, \*, /, %, =, !=, >, >=, <, and <= comprise a set of arithmetic or logical operators for constructing one or more expression for use with an if construct selectively to make one or more connection in a for loop.

40-52. (Cancelled)